UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/722,296 | 11/25/2003 | Alnasir Ladha | 11312 | 9107 |

7590 07/17/2007

James M. Stover
NCR Corporation
1700 South Patterson Blvd.
Dayton, OH 45479-0001

| EXAMINER | |
|---|---|
| KUDDUS, DANIEL A | |
| ART UNIT | PAPER NUMBER |
| 2169 | |

| MAIL DATE | DELIVERY MODE |
|---|---|
| 07/17/2007 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

| Office Action Summary | Application No. | Applicant(s) | |
|---|---|---|---|
| | 10/722,296 | LADHA ET AL. | |
| | Examiner | Art Unit | |
| | Daniel Kuddus | 2169 | |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *11 April 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *8-25* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *8-25* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *11/25/03* is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

# DETAILED ACTION

## Response to Amendment

This Office action has been issued in response to amendment filed April 18, 2007. Claims 8-25 are pending. Applicant's arguments are carefully and respectfully considered and some are persuasive, while others are not. Accordingly rejections have been removed where arguments were persuasive, but rejections have been maintained where arguments were not persuasive. Also, new rejections based on the newly added claims have been set forth. Accordingly, claims 8-25 are rejected and this action has been made FINAL, as necessitated by amendment.

## Response to Arguments

With respect to applicant's arguments that " It is not executed in parallel and there are not multiple instances of the application process 60 on the network" and "different aspects of the application process 60 or operations are distributed by the "application process" itself is not in fact duplicated and does not appear as "multiple instances" on one or more nodes", the examiner disagree with applicant's, because Klein reference teaches on column 5, lines 51-67, various tables within a database may be stored on different nodes of the system. Such distributed storage facilitates efficient, 'parallel' processing of queries, by 'distributing' both the disk I/O and computational burden over multiple nodes. The leaf nodes are executed by disk processes in each of the nodes of the transaction processing system. The number of disk processes per node may vary from one implementation to another. A separate disk processes may be used for each logical disk volume.

With respect to applicant's arguments that "multiple or one or more instances of applications each instance processing in parallel on other nodes" and "multiple instances over the network" these limitation is not in Klein. The examiner disagree with applicant's arguments. Klein reference does teach these limitations, as noted above on column 5, line 51-67, note that when parallel processing of queries distributing over multiple nodes and disks, and leaf nodes are executed by the dick processes in each of the nodes of the transactional processing system, there are multiple instances of applications each instance processing in parallel on the nodes and over the network. Additionally Klein teaches on column 14, line 42-58, transparent horizontal partitioning of large database tables is used to scale and distribute computational loads across the nodes and devices in a cluster. "Horizontal partitioning" means that a table is divided or partitioned into two or more files, with each partition storing the records (tuples) having a corresponding range of key values. Each partition may be stored on a different node of the system to facilitate distribution of the computational load on the system, which reads the claim language. Also see column 6, line 57-62, a network or other communication interface for communicating with other computers. One or more communication buses for interconnecting the CPU(s), memory, user interface and network interface, and column 22, line 44-48, the software modules in the computer program product also be distributed electronically, via the Internet or otherwise, by transmission of a computer data signal (in which the software modules are embedded) on a carrier wave.

Therefore the examiner conclude teaching of Klein reads on the claim invention and the claim invention is not distinct over Klein.

## Claim Rejection Under 35 U.S.C. § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 8-25 are rejected under 35 U.S.C. 102(b) as being anticipated by Klein et al. ('Klein' hereinafter), US 6,453,313 B1.

With respect to claim 8,

Klein teaches a method to manage interactions between applications and a data store (see Figs. 6, 15), comprising:

receiving a query for a data store and an identifier for an application, wherein the application when executed seeks to process results returned from and produced by executing the query and seeks to update the data store with application data, wherein the application data is produced in response to the application processing the results of the query (The fan out operator sends this request 'query' to each table partition, and receives in response all records that satisfy the cursor. The request is non-blocking because the fan out operator does not want or need to receive records added 'update' to the table partition after the request is made. This type of request is used for streaming, read only access (i.e., for streaming operators that do not delete or update tuples). This type of request is sent by the fan out operator to all of the partition scan operators so as to automatically retrieve rows as they are inserted or updated in the table. The delete and update

features of the present invention provide a destructive read capability and a "read modify write" capability in conjunction with streaming access to a database table. This allows queuing services to be provided by a relational database system while preserving the ability of the DBMS to perform other relational operators on the result set returned. The result sets created by the delete and update access operations of the present invention can be joined with the result sets of other table access operators, which enables efficient data processing through the use of delete and/or update operations embedded in a query, see col.15, lines 48-60, column 16, lines 66 to column 17, lines 6, column 18, line 12-15, Figs.15, 18, Klein),

concurrently executing initiating multiple instances of an application associated with the identifier on multiple processing nodes within a network to achieve parallel processing for the multiple instances of the application (tables in the database are partitioned, with various partitions being stored on different nodes of the relational database system. Such partitioning is often used for extremely large tables. Various tables within a database are stored on different nodes of the system. Such distributed storage facilitates efficient, parallel 'concurrent' processing of queries, by distributing both the disk I/O and computational burden over multiple nodes. The "application process" represents the process or processes that execute not only the application program, but also the portions of the execution tree above the leaf nodes. The leaf nodes are executed by disk processes in each of the nodes of the transaction processing system. While one disk process for each node, the number of disk processes per node may vary from one implementation to another. A separate disk process may be used for each logical disk volume. Destructive reads are sometimes used to ensure that an item is processed exactly once. For

instance, several "credit evaluation" processes might be assigned the job of reading and processing credit applications. Each such process could use a destructive read (i.e., delete operation with result set) to read a next credit application record for processing. The credit evaluation processes work in parallel, without interfering with each other, see col. 5, lines 51-67, column 6, line 57-62, column 14, line 42-58, column 17, line 9-16, Klein).

concurrently processing the query and producing housing the results that are then housed in one or more application queues residing on one or more of the processing nodes (data flows between the nodes of the execution tree are handled by the use of a pair of queues, between parent and child nodes. In particular each parent node is coupled to a child node by a request queue and a fetched records queue. The request queue stores requests being conveyed from the parent node to its child node, while the fetched records queue conveys data and return codes (e.g., an end of file or end of scan code) being returned to the parent node in response to the requests (see col. 6, lines 1-10, Klein).

concurrently providing the results to each of the instances of the application from the one or more application queues so that the instances can produce the application data from the results and update the data store with the application data, which is to be subsequently accessed from the data store (the request queue and a fetched records queue are used by the transaction processing system to pre-fetch records not yet requested by the application that submitted the query being processed. Each node in the execution tree other than the leaf nodes are automatically configured to request as many records as can be stored in the fetched records queue(s) between it and its child or children nodes, even if such records have not yet been

requested by the application. Pre-fetching can improve system performance, by making use of otherwise dormant system resources, and can improve system responsiveness by having data ready for the application before it requests it, unbound pre-fetching must be suppressed when executing an embedded delete or update statement. The application must control how many rows are to be affected by the delete or update operation, and therefore the database management system must only delete or update those records actually requested by the application, SQL compiler includes in the code for any update, delete or insert operator (generically herein called a table access operator) code for generating a before and after image for each modified and new tuple. SQL compiler of the present invention the image generation code includes code for updating one or more fields of the Before Image when the query being compiled includes a "set on rollback" clause that affects the table being accessed by this operator. When the Before and After Images are passed by the table access operator to the transaction log manager, the before image contains one or more modified fields if the query being executed contained a corresponding "set on rollback" clause, see col. 13, lines 64 to col. 14, lines 6, column 19, line 17-35, Klein).

As to claim 9,

Klein teaches concurrently housing the application data in one or more load queues residing on one or more of the processing nodes (see col. 14, lines 59-63, Klein); and concurrently populating one or more tables residing on one or more of the processing nodes with the application data from the one or more load queues (see col. 14, lines 59-63, Klein).

As to claim 10,

Klein teaches merging the one or more tables into the data store (see col. 3, lines 15-18, Klein).

As to claim 11,

Klein teaches wherein the currently initiating further includes determining a total number of the applications to initiate based on configuration data (see col. 6, lines 1-10, Klein).

As to claim 12,

Klein teaches wherein the currently initiating further includes determining which of a number of the applications that are to be initiated on which of a number of the processing nodes based on the configuration data (see col. 14, lines 10-15, Klein).

As to claim 13,

Klein teaches concurrently synchronizing the application queues and the load queues on the multiple processing nodes when at least some of the processing nodes lack one of the one or more application queues or one of the one or more load queues (see col. 14, lines 59-63, Klein).

As to claim 14,

Klein teaches wherein the concurrently synchronizing further includes establishing socket based communications between the multiple processing nodes with a Transmission ControlProtocol/Internet Protocol (TCP/IP) (see col. 19, lines 30-35, Klein).

Claims 15-25 have the same subject matter except temporary tables, data warehouse and query results extraction and Klein teaches at col. 11, lines 11-17, lines 47-55, col. 15, lines 26-35, lines 66 to column 16, line 1-11, and line 58-65 et seq., and essentially rejected for the same reasons as discussed above.

## Conclusion

The prior art made of record and not relied upon is considered pertinent toapplicant's disclosure. Kerwin et al. (USP, 6,898,609) teaches all the limitations especially "provides a software method, for network database environments, permitting load balancing, scalability and substantially simultaneous use by client users, comprising the steps of: providing multiple database instances wherein each such instance is substantially identical in data content, database structure, and primary key system; maintaining substantially real time records of status for each such multiple database instance; receiving a database query from at least one end-user application and determining such query to be a transactional query or non-transactional query; directing such database query to at least one selected instance of such multiple database instances upon a determination of such query being a non-transactional query; returning such non-transactional query results to the at least one end-user application; directing such database query to all instances of such multiple database instances upon a determination of such query being a transactional query; controlling such transactional queries to maintain substantial identicalness among such multiple database instances; propagating such transactional queries to such multiple database instances; returning such query results to the user; recognizing a failure in at least one instance of such multiple database instances, and adjusting to store such transactional query for later propagation; restoring such failed at least one instance of such multiple database instances to substantial identicalness with other such multiple database instances. Moreover, it provides such a method wherein each such non-transactional query is executed upon a randomly selected instance of such multiple database instances. Additionally, it provides such a method wherein the

processing of such non-transactional query commands as directed by a plurality of users is substantially simultaneous" see col. 5, lines 1-25, and col. 7, lines 1-16, Kerwin et al.

## Conclusion

**THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

## Contact Information

Any inquiry concerning this communication or earlier communication from the examiner should be directed to Daniel A Kuddus whose telephone number is (571) 270-1722. The examiner can normally be reached on Monday to Thursday 8.00 a.m.-5.30 p.m. The examiner can also be reached on alternate Fridays from 8.00 a.m. to 4.30 p.m.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor Pierre

M. Vital can be reached on (571) 272-4215. The fax phone number for the organization where this application or processing is assigned is 571-273-8300. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from the either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Daniel Kuddus

Date: 07/03/07

/CDL/

PIERRE VITAL
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100